

Narration	Time
++ C میں Exception Handling کے سپوکن ٹیوٹورل میں خوش آمدید۔	00:01
اس ٹیوٹورل میں ہم Exception Handling سیکھیں گے۔	00:07
ہم یہ ایک مثال کی مدد سے کریں گے۔	00:11
اس ٹیوٹورل کو ریکارڈ کرنے کے لئے، میں استعمال کارہا ہوں،	00:14
Ubuntu OS ورژن 11.10	00:16
++ g کمپائلر ورژن 4.6.1	00:20
اب 'Exception' کے تعارف کے ساتھ شروع کرتے ہیں۔	00:25
'Exception' ایک مسئلہ ہے جو پروگرام کے عملدرآمد کے وقت پیدا ہوتا ہے۔	00:29
یہ رن-ٹائم ایرر ہے جس کا پتہ پروگرام لگا سکتا ہے۔	00:34
اب ایکسپشن ہینڈلنگ پر جاتے ہیں۔	00:39
پروگرام کے عملدرآمد کے وقت مسائل کو دئے گئے رد عمل کو ایکسپشن ہینڈلنگ کہتے ہیں۔	00:42
ایکسپشن ہینڈلنگ کارکردگی کو جاری رکھنے کی اجازت دیتا ہے۔	00:50
یہ مسئلہ کا پتہ لگانے میں مدد کرتا ہے،	00:55
اور کنٹرول طریقے سے پروگرام کو ختم کرتا ہے۔	00:57
اب ایکسپشن کی قسمیں دیکھتے ہیں	01:02
'Try'	01:05
'Catch'	01:06
اور 'Throw'	01:07
ہم ایرر پر ون کوڈ کو try block کے اندر رکھتے ہیں۔	01:09
پھر اسے throw استعمال کرتے ہوئے سمبھالا جاتا ہے۔	01:13
اس کے بعد catch اسٹیٹمنٹ کرتے ہوئے اس ایکسپشن کو پکڑا جاتا ہے۔	01:16
اور پھر یہ عمل آ رہوتا ہے۔	01:21
try، catch اور throw کے لئے سٹیکس یعنی ساخت ہے	01:23

Throw:	01:27
catch block اور try block	01:28
یہاں ہم آرگیمینٹ پاس کرتے ہیں۔	01:32
Throw اسٹیٹمنٹ try block کے اندر بھی لکھا جاسکتا ہے۔	01:35
ہمارے پاس ایک سے زیادہ try blocks، catch blocks ہیں۔	01:40
اب ایکسپشن ہینڈلنگ پر ایک مثال دیکھتے ہیں۔	01:44
میرے پاس کوڈ ہے، میں اسے کھولوں گا۔	01:48
نوٹ کریں، ہمارا فائل نیم 'exception.cpp' ہے۔	01:51
اس پروگرام میں ہم ایکسپشن ہینڈلنگ استعمال کر کے صفر سے تقسیم والا ایرر حل کریں گے۔	01:55
اب پورے کوڈ کو دیکھتے ہیں۔	02:02
یہ 'iostream' ہماری ہیڈر فائل ہے۔	02:04
یہاں ہم 'std namespace' استعمال کر رہے ہیں۔	02:07
یہاں ہمارے پاس آرگیمینٹ 'int a' اور 'int b' کے ساتھ فنکشن 'division' ہے۔	02:11
پھر ہم جانچتے ہیں کہ 'b == 0' ہے یا نہیں	02:18
اگر ٹرو ہے، تو ہم division by zero condition ایکسپشن کو throw کرتے ہیں۔	02:22
فنکشن 'a' اور 'b' کے ڈویژن کو رٹرن کرتا ہے۔	02:27
یہ ہمارا main فنکشن ہے۔	02:32
اس میں ہم انٹر ویریبل 'x'، 'y' اور ایک ڈبل ویریبل 'z' ڈیکلئر کرتے ہیں۔	02:34
یہاں ہم 'x' اور 'y' کی ویلیو لیتے ہیں۔	02:42
یہ ہمارا 'try block' ہے۔	02:46
یہاں ہم نے فنکشن 'division' کال کیا۔	02:48
اور نتیجے کو 'z' میں اسٹور کیا ہے۔	02:51
پھر ہم 'z' کی ویلیو پرنٹ کرتے ہیں۔	02:54
یہ ہمارا 'catch block' ہے۔	02:57

02:59	اس میں ہم ایک 'argument msg'، 'character constant' کو طور پر پاس کرتے ہیں۔
03:06	پھر ہم 'msg' پرنٹ کرتے ہیں۔
03:08	اور یہ ہمارا return اسٹیٹمنٹ ہے۔
03:11	اب پروگرام اکیز کیوٹ کرتے ہیں۔
03:13	اپنے کی بورڈ پر ایک ساتھ Ctrl، Alt اور T کیز دبا کر ٹرمینل ونڈو کھولیں۔
03:21	کمپائل کرنے کے لئے، ٹائپ کریں g++ space exception dot cpp space hyphen o space ex.
03:32	ٹائپ کریں dot slash ex,
03:36	x اور y کی ویلیو اینٹر کریں۔
03:38	میں 3 اور 0 دوں گا۔
03:42	آؤٹ پٹ ظاہر ہوتا ہے Division by zero condition
03:46	دوبارہ کمپائل کرتے ہیں۔
03:48	اپ ایرو کی (key) دوبار دبا ئیں۔
03:51	اینٹر دبا ئیں۔
03:52	دوبارہ اپ ایرو کی (key) دوبار دبا ئیں۔
03:55	x اور y کی ویلیو اینٹر کریں۔
03:57	میں 8 اور 2 دوں گا۔
04:01	آؤٹ پٹ 4 ہے۔
04:04	اس طرح try, catch اور throw block کام کرتے ہیں۔
04:08	اسی کے ساتھ ہم اس ٹیوٹورل کے آخر میں آگئے ہیں
04:11	اب اپنی سلائڈ پر واپس آتے ہیں۔
04:14	اور خلاصہ کرتے ہیں۔
04:16	اس ٹیوٹورل میں ہم نے دیکھا 'Exception Handling'، 'Try' 'Catch' اور 'Throw blocks'

04:23	ایک مشق ملازمین کی ایجنسی عمر ظاہر کریں۔
04:26	ایک ایکسپشن تھرو کریں یہ جانچنے کے لئے کہ ایجنسی 15 سے کم نہیں ہے۔
04:31	مندرجہ ذیل لنک پر دستیاب وڈو دیکھیں۔
04:34	یہ سپوکن ٹیوٹوریل پروجیکٹ کو خلاصہ کرتا ہے۔
04:38	اگر آپ کے پاس اچھی بینڈ وڈتھ نہ ہو تو آپ اسے ڈاؤن لوڈ کر کے دیکھ سکتے ہیں
04:42	سپوکن ٹیوٹوریل پروجیکٹ ٹیم، سپوکن ٹیوٹوریلز کا استعمال کرتے ہوئے ورکشاپس چلاتی ہے۔
04:48	اور آن لائن ٹیسٹ پاس کرنے والوں کو سہارا دیتے ہیں۔
04:52	مزید معلومات دے لئے contact@spoken-tutorial.org کو لکھیں۔
04:59	سپوکن ٹیوٹوریل پروجیکٹ ٹاک-ٹو-اے-ٹیچر پروجیکٹ کا حصہ ہے۔
05:04	اسے ایم ایچ آر ڈی، حکومت ہند کے آئی سی ٹی کے ذریعے قومی خواندگی مشن کی طرف سے حمایت حاصل ہے۔
05:11	اس مشن پر مزید معلومات درج ذیل لنک پر دستیاب ہے : http://spoken-tutorial.org\NMEICT-Intro
05:16	اس اسکرپٹ کا ترجمہ اور صدا بندی میں نے یعنی وجاحت احمد نے کی ہے، ہمارے ساتھ شریک ہونے کیلئے بہت شکریہ